

## A VARIANT OF THE AOR METHOD FOR AUGMENTED SYSTEMS

M. M. MARTINS, W. YOUSIF, AND J. L. SANTOS

**ABSTRACT.** In this paper we present a variant of the Accelerated Overrelaxation iterative method (AOR), denoted by modified AOR-like method (MAOR-like method) for solving the augmented systems, i.e., the AOR-like method with three real parameters  $\omega$ ,  $r$  and  $\alpha$ . For special values of  $\omega$ ,  $r$  and  $\alpha$  we get the MSOR-like method, the AOR-like method and the SOR-like method. An equation relating the involved parameters and the eigenvalues of the iteration matrix of the MAOR-like method is obtained. Furthermore, some convergence conditions for the MAOR-like method are derived. This paper generalizes the main results of Li, Li, Nie, and Evans 2004 and Shao, Li, and Li (2007). Numerical examples are presented to show that, for a suitable choice of the involved parameters, the MAOR-like method is superior when compared to the above iterative methods and to the SSOR-like method presented by Zheng, Wang, and Wu (2009).

### 1. INTRODUCTION AND PRELIMINARIES

Many applications such as fluid dynamics, optimization and constrained or generalized least squares problems, image processing, linear elasticity and mixed Finite Element Method for elliptic equations [1]–[3], [6] and [15] lead us to a linear system of equations of the form:

$$(1.1) \quad \begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} p \\ q \end{pmatrix},$$

where  $A \in \mathbb{R}^{m,m}$  is a symmetric and positive definite (SPD) matrix,  $B \in \mathbb{R}^{m,n}$  is a matrix which has a full column rank, i.e.,  $\text{rank}(B) = n$ ,  $0 \in \mathbb{R}^{n,n}$  and obviously  $m \geq n$ , the vectors  $x, p \in \mathbb{R}^m$  and the vectors  $y, q \in \mathbb{R}^n$ ,  $B^T$  denotes the transpose of the matrix  $B$ .

In this paper, the approximation to the solution of the linear system of equations (1.1), which, under the above conditions, has a unique solution, will be obtained. The iterative methods are normally used to obtain an approximation of the solution of (1.1) because of storage requirements and since these methods will not change the structure of the original matrix and therefore preserve sparsity. The well-known Accelerated Overrelaxation iterative (AOR) method introduced by Hadjidimos in [5] can be used if the matrix  $A$  of the system (1.1) is nonsingular, many authors have improved the convergence of the AOR method, for instance, [12] and [13].

---

Received by the editor August 7, 2009 and, in revised form, September 20, 2010.

2010 *Mathematics Subject Classification.* Primary 65F10.

*Key words and phrases.* AOR-like method, SOR-like method, MAOR-like method, MSOR-like method, SSOR-like method, saddle point problem, augmented system, iterative methods.

Recently, several authors [7]–[11] and [14] have been generalizing the Successive Overrelaxation iterative (SOR) method for the system (1.1).

The most practical and important generalization of the SOR method is the SOR-like method given by Golub et al. [4]. The splitting related to the SOR-like method is

$$(1.2) \quad \begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} = \begin{pmatrix} A & 0 \\ 0 & Q \end{pmatrix} - \begin{pmatrix} 0 & 0 \\ -B^T & 0 \end{pmatrix} - \begin{pmatrix} 0 & -B \\ 0 & Q \end{pmatrix} = D - L - U,$$

where  $Q \in \mathbb{R}^{n,n}$  being the preconditioning matrix and the SOR-like method is defined by the normal SOR procedure:

$$(1.3a) \quad (D - \omega L) \begin{pmatrix} x^{(k+1)} \\ y^{(k+1)} \end{pmatrix} = \omega \begin{pmatrix} p \\ q \end{pmatrix} + [(1 - \omega)D + \omega U] \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix}, \quad k = 0, 1, 2, \dots$$

with

$$(1.3b) \quad D = \begin{pmatrix} A & 0 \\ 0 & Q \end{pmatrix}, L = \begin{pmatrix} 0 & 0 \\ -B^T & 0 \end{pmatrix}, \text{ and } U = \begin{pmatrix} 0 & -B \\ 0 & Q \end{pmatrix}.$$

The corresponding AOR-like method [11], denoted in that paper, as generalized AOR (GAOR) is defined as follows:

$$(1.4) \quad (D - rL) \begin{pmatrix} x^{(k+1)} \\ y^{(k+1)} \end{pmatrix} = \omega \begin{pmatrix} p \\ q \end{pmatrix} + [(1 - \omega)D + (\omega - r)L + \omega U] \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix}, \quad k = 0, 1, 2, \dots$$

where  $\omega$  and  $r$  are real numbers.

In this paper, the AOR-like method is modified where an extra parameter  $\alpha$  is introduced. Let us consider the splitting

$$(1.5) \quad \begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} = \begin{pmatrix} A & 0 \\ 0 & Q \end{pmatrix} - \begin{pmatrix} 0 & 0 \\ -B^T & \alpha Q \end{pmatrix} - \begin{pmatrix} 0 & -B \\ 0 & \beta Q \end{pmatrix},$$

where  $Q$  is a nonsingular and symmetric matrix and  $\alpha + \beta = 1$ .

If we denote

$$(1.6) \quad D_1 = \begin{pmatrix} A & 0 \\ 0 & Q \end{pmatrix}, L_1 = \begin{pmatrix} 0 & 0 \\ -B^T & \alpha Q \end{pmatrix}, \text{ and } U_1 = \begin{pmatrix} 0 & -B \\ 0 & \beta Q \end{pmatrix},$$

then a variant of the AOR-like method, denoted by the modified AOR-like method with three parameters can be defined as

$$(1.7) \quad (D_1 - rL_1) \begin{pmatrix} x^{(k+1)} \\ y^{(k+1)} \end{pmatrix} = \omega \begin{pmatrix} p \\ q \end{pmatrix} + [(1 - \omega)D_1 + (\omega - r)L_1 + \omega U_1] \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix}, \quad k = 0, 1, 2, \dots$$

The above modified AOR-like (MAOR-like) method involves three parameters  $\omega$ ,  $r$  and  $\alpha$ , and one preconditioning matrix  $Q$ . If the parameter  $\alpha = 0$ , i.e.  $\beta = 1$ , then the MAOR-like method becomes the AOR-like method; in addition, if  $\omega = r$ , then we have the SOR-like method given by Golub et al. [4].

Furthermore, we will establish some convergence conditions for the MAOR-like method, which generalizes those presented in [11] and [14].

From (1.6), we have

$$(1.8) \quad (D_1 - rL_1) = \begin{pmatrix} A & 0 \\ rB^T & (1 - r\alpha)Q \end{pmatrix},$$

where  $A$  is a SPD matrix and the matrix  $Q$  is nonsingular, therefore we have

$$\det(D_1 - rL_1) = (1 - \alpha r)^n \det(A) \det(Q) \neq 0$$

if and only if

$$(1.9) \quad 1 - \alpha r \neq 0 \quad \text{or, equivalently,} \quad \alpha r \neq 1.$$

The analysis of the convergence of the MAOR-like method is presented in Sections 2 and 3. Based on a functional equation involving the parameters  $\omega$ ,  $r$  and  $\alpha$ , and the eigenvalues of the MAOR-like iteration matrix and those of the matrix  $-Q^{-1}B^T A^{-1}B$ , we will also obtain the convergence region for this iterative method in Section 3. Then in order to compare the performance of the MAOR-like method to the other methods, numerical experiments have been carried out and the results are summarized in Section 4. The results obtained show that with a suitable choice of the parameters  $\omega$ ,  $r$  and  $\alpha$  the MAOR-like method is computationally competitive in comparison with the iterative methods presented earlier in this section.

Recently, Zheng et al. [17] have proposed the SSOR-like method to approximate the solution of (1.1), where different choices of the matrix  $Q$  were considered. One of the numerical examples they have presented was the Stokes problem. A comparison between the results obtained with the MAOR-like method and those obtained with SSOR-like method of [17], for this problem, is given in Section 4. The numerical results show that the MAOR-like method is more efficient than the SSOR-like method of [17].

The SSOR-like method, for the splitting (1.2), is given by

$$(1.10) \quad \begin{pmatrix} x^{(k+1)} \\ y^{(k+1)} \end{pmatrix} = U_\omega L_\omega \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix} + \omega(2 - \omega)(D - \omega U)^{-1} D (D - \omega L)^{-1} \begin{pmatrix} p \\ q \end{pmatrix}, \quad k = 0, 1, 2, \dots$$

where

$$\begin{aligned} L_\omega &= (D - \omega L)^{-1} [(1 - \omega)D + \omega U], \\ U_\omega &= (D - \omega U)^{-1} [(1 - \omega)D + \omega L], \end{aligned}$$

with the matrices  $D$ ,  $L$  and  $U$  defined in (1.3b).

## 2. SOME RESULTS AND THE BASIC FUNCTIONAL EQUATION

Let  $L_{r,\omega,\alpha}$  be the iteration matrix of the MAOR-like method. From (1.8) we can write

$$(D_1 - rL_1)^{-1} = \begin{pmatrix} A^{-1} & 0 \\ -\frac{r}{1-r\alpha} Q^{-1} B^T A^{-1} & \frac{1}{1-r\alpha} Q^{-1} \end{pmatrix}.$$

Since  $L_{r,\omega,\alpha} = (D_1 - rL_1)^{-1}[(1 - \omega)D_1 + (\omega - r)L_1 + \omega U_1]$ , then

$$L_{r,\omega,\alpha} = \begin{pmatrix} A^{-1} & 0 \\ -\frac{r}{1-r\alpha}Q^{-1}B^T A^{-1} & \frac{1}{1-r\alpha}Q^{-1} \end{pmatrix} \times \begin{pmatrix} (1-\omega)A & -\omega B \\ (r-\omega)B^T & (1-\omega)Q + (\omega-r)\alpha Q + \omega\beta Q \end{pmatrix}$$

but, as  $\alpha + \beta = 1$ , we have

$$L_{r,\omega,\alpha} = \begin{pmatrix} A^{-1} & 0 \\ -\frac{r}{1-r\alpha}Q^{-1}B^T A^{-1} & \frac{1}{1-r\alpha}Q^{-1} \end{pmatrix} \begin{pmatrix} (1-\omega)A & -\omega B \\ (r-\omega)B^T & (1-r\alpha)Q \end{pmatrix}$$

or, equivalently,

$$(2.1) \quad L_{r,\omega,\alpha} = \begin{pmatrix} (1-\omega)I_m & -\omega A^{-1}B \\ \frac{\omega(r-1)}{1-r\alpha}Q^{-1}B^T & I_n + \frac{\omega r}{1-r\alpha}Q^{-1}B^T A^{-1}B \end{pmatrix},$$

where  $\alpha$  and  $r$  satisfy (1.9).

Note that if  $\omega = 0$ , then

$$L_{r,0,\alpha} = \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix}.$$

Therefore the MAOR-like method diverges no matter what values the parameters  $\alpha$  and  $r$  take. Thus, in what follows, we will assume that the parameter  $\omega$  is different from zero, with  $\alpha$  and  $r$  satisfying (1.9).

In order to get the functional equation let  $\lambda$  be an eigenvalue of  $L_{r,\omega,\alpha}$  and  $(u^T, v^T)^T$  be the corresponding eigenvector, then we have

$$(2.2) \quad L_{r,\omega,\alpha} \begin{pmatrix} u \\ v \end{pmatrix} = \lambda \begin{pmatrix} u \\ v \end{pmatrix}.$$

From (2.1), we can write (2.2) as

$$\begin{cases} (1-\omega)u - \omega A^{-1}Bv = \lambda u, \\ \frac{\omega(r-1)}{1-r\alpha}Q^{-1}B^T u + v + \frac{\omega r}{1-r\alpha}Q^{-1}B^T A^{-1}Bv = \lambda v \end{cases}$$

or, equivalently,

$$\begin{cases} (1-\omega-\lambda)u = \omega A^{-1}Bv, \\ \omega(r-1)Q^{-1}B^T u + r(1-\omega-\lambda)Q^{-1}B^T u = (1-\lambda)(1-r\alpha)v \end{cases}$$

which is equivalent to

$$(2.3) \quad \begin{cases} (1-\omega-\lambda)u = \omega A^{-1}Bv, \\ (r\lambda + \omega - r)Q^{-1}B^T u = (1-\lambda)(1-r\alpha)v. \end{cases}$$

From (2.3) we get the following result:

**Lemma 2.1.** *If  $\lambda$  is an eigenvalue of  $L_{r,\omega,\alpha}$ , then  $\lambda \neq 1$ .*

*Proof.* The proof is similar to the one given in [11].  $\square$

**Lemma 2.2.** *If  $r = 1$ , then  $\lambda = 1 - \omega$  is an eigenvalue of  $L_{r,\omega,\alpha}$  with multiplicity of  $m$ . If  $r \neq 1$ , then  $\lambda = 1 - \omega$  is an eigenvalue of  $L_{r,\omega,\alpha}$  if and only if  $m > n$ ; in this case the multiplicity of  $\lambda$  is  $m - n$ .*

*Proof.* If  $\lambda = 1 - \omega$  is an eigenvalue of  $L_{r,\omega,\alpha}$ , then there exists a nonzero vector  $(u^T, v^T)^T$  so that they satisfy the system (2.2). Therefore, from (2.3) and since  $\omega \neq 0$  we have

$$(2.4) \quad \begin{aligned} A^{-1}Bv &= 0 \\ (1-r)Q^{-1}B^T u &= (1-r\alpha)v. \end{aligned}$$

Since the matrix  $B$  has a full column rank, then the above system is equivalent to

$$v = 0 \quad \text{and} \quad (1-r)Q^{-1}B^T u = 0.$$

As  $B^T$  is a matrix of order  $n \times m$  and  $\text{rank}(B^T) = n$ , we will consider the following cases:

- (i) If  $r = 1$  we have  $(1-r)Q^{-1}B^T u = 0$  true for any vector  $u \in \mathbb{R}^m$ . Therefore  $\lambda = 1 - \omega$  is an eigenvalue of  $L_{r,\omega,\alpha}$  of multiplicity  $m$ .
- (ii) If  $r \neq 1$  and  $m > n$ , we have  $v = 0$  and  $Q^{-1}B^T u = 0$ . Thus,  $\lambda = 1 - \omega$  is an eigenvalue of  $L_{r,\omega,\alpha}$  with multiplicity of  $m - n$ , as the linear system  $B^T u = 0$  has  $m - n$  linearly independent solutions.
- (iii) If  $r \neq 1$  and  $m = n$ , then  $Q^{-1}B^T u = 0$  has only a null solution. Therefore,  $\lambda = 1 - \omega$  is not an eigenvalue of  $L_{r,\omega,\alpha}$  and we get the required result.  $\square$

*Remarks.* (1) If we let  $\omega = r$ , in Lemma 2.1 and Lemma 2.2, then we obtain Lemma 1 and Lemma 2 of [14], respectively.

- (2) If  $\alpha = 0$ , then Lemma 2.1 and Lemma 2.2 become Lemma 1 and Lemma 2 of [11], respectively.
- (3) If  $\lambda$  and  $(u^T, v^T)^T$  are the eigenvalue and eigenvector of  $L_{r,\omega,\alpha}$ , respectively, and if  $\lambda \neq 1 - \omega$ , then  $v \neq 0$ .

**Theorem 2.1.** *Let  $L_{r,\omega,\alpha}$  be the iteration matrix of the MAOR-like method, with the parameters  $r$  and  $\alpha$  satisfying (1.9) and  $\omega \neq 0$  then:*

- (1)  $\lambda = 1 - \omega$  is an eigenvalue of  $L_{r,\omega,\alpha}$  if  $m > n$ .
- (2) For any eigenvalue  $\lambda (\neq 1 - \omega)$  of  $L_{r,\omega,\alpha}$  there is an eigenvalue  $\mu$  of  $-Q^{-1}B^T A^{-1}B$ , so that  $\lambda, \mu, r$  and  $\omega$  satisfy the following functional equation:

$$(2.5) \quad (1 - \omega - \lambda)(1 - \lambda)(1 - r\alpha) = \omega(r - \omega - r\lambda)\mu.$$

- (3) For any eigenvalue  $\mu$  of  $-Q^{-1}B^T A^{-1}B$ , if  $\lambda$  is different from  $1 - \omega$  and from 1, and if  $\mu, \alpha, r$  and  $\omega$  satisfy equation (2.5), then  $\lambda$  is an eigenvalue of  $L_{r,\omega,\alpha}$ .

*Proof.* From Lemma 2.2 we obtain conclusion 1.

In order to prove the equality (2.5), let us consider the eigenvalue  $\lambda (\neq 1 - \omega)$  of  $L_{r,\omega,\alpha}$  and the corresponding eigenvector  $(u^T, v^T)^T$ , which satisfy (2.3), the first equality of (2.3) can be written as

$$(2.6) \quad u = \frac{\omega}{1 - \omega - \lambda} A^{-1}Bv.$$

Hence, we can rewrite the second equality of (2.3) as

$$(2.7) \quad (1 - \omega - \lambda)(1 - \lambda)(1 - r\alpha)v = (r\lambda\omega + \omega^2 - r\omega)Q^{-1}B^T A^{-1}Bv.$$

Since  $\lambda \neq 1 - \omega$ ,  $\lambda \neq 1$  (by Lemma 2.1),  $v \neq 0$  (from Remark (3). after Lemma 2.2) and  $1 - r\alpha \neq 0$  (see (1.9)), then  $r\lambda\omega + \omega^2 - r\omega \neq 0$ . Thus, there is an eigenvalue  $\mu$

of  $-Q^{-1}B^T A^{-1}B$  so that  $\lambda$ ,  $\mu$  and  $\omega$  satisfy equation (2.5). Therefore, the proof of the second part is concluded.

For the last part of the theorem, let  $\mu$ ,  $v$  ( $\neq 0$ ) be an eigenvalue and eigenvector of the matrix  $-Q^{-1}B^T A^{-1}B$ , respectively, therefore we have  $-Q^{-1}B^T A^{-1}Bv = \mu v$ . By the conditions of the theorem, equation (2.7) holds, since  $\mu \neq 0$ , as  $-Q^{-1}B^T A^{-1}B$  is a nonsingular matrix. Also, from equation (2.6), we can rewrite equation (2.7) as follows:

$$(1 - \lambda)(1 - r\alpha)v = (r\lambda + \omega - r)Q^{-1}B^T u.$$

Thus, the system (2.3) holds, which is equivalent to the system (2.2). Therefore,  $\lambda$  is an eigenvalue of  $L_{r,\omega,\alpha}$ , as the associated eigenvector  $(u^T, v^T)^T$  is nonnull, resulting in part (3) of the theorem.  $\square$

**Corollary 2.1.** *Let  $L_{\omega,\omega,\alpha}$  be the iteration matrix of the MSOR-like method, then:*

- (1)  $\lambda = 1 - \omega$  is an eigenvalue of  $L_{\omega,\omega,\alpha}$  if  $m > n$ .
- (2) For any eigenvalue  $\lambda$  ( $\neq 1 - \omega$ ) of  $L_{\omega,\omega,\alpha}$  there is an eigenvalue  $\mu$  of  $-Q^{-1}B^T A^{-1}B$ , so that  $\lambda$ ,  $\mu$  and  $\omega$  satisfy the following functional equation:

$$(2.8) \quad (1 - \omega - \lambda)(1 - \lambda)(1 - \omega\alpha) = \lambda\omega^2\mu.$$

- (3) For any eigenvalue  $\mu$  of  $-Q^{-1}B^T A^{-1}B$ , if  $\lambda$  is different from  $1 - \omega$  and from 1, and if  $\alpha$ ,  $\mu$  and  $\omega$  satisfy equation (2.8), then  $\lambda$  is an eigenvalue of  $L_{\omega,\omega,\alpha}$ .

**Corollary 2.2.** *Let  $\rho(L_{r,\omega,\alpha})$  be the spectral radius of the MAOR-like iteration matrix  $L_{r,\omega,\alpha}$  and  $m > n$ , then we have*

$$\rho(L_{r,\omega,\alpha}) \geq |1 - \omega|.$$

**Corollary 2.3.** *Let  $L_{r,\omega,0}$  be the iteration matrix of the GAOR (AOR-like) method, defined in (1.4), then:*

- (1)  $\lambda = 1 - \omega$  is an eigenvalue of  $L_{r,\omega,0}$  if  $m > n$ .
- (2) For any eigenvalue  $\lambda$  ( $\neq 1 - \omega$ ) of  $L_{r,\omega,0}$  there is an eigenvalue  $\mu$  of  $-Q^{-1}B^T A^{-1}B$ , so that  $\lambda$ ,  $\mu$  and  $\omega$  satisfy the following functional equation:

$$(2.9) \quad (1 - \omega - \lambda)(1 - \lambda) = \omega(r - \omega - r\lambda)\mu.$$

- (3) For any eigenvalue  $\mu$  of  $-Q^{-1}B^T A^{-1}B$ , if  $\lambda$  is different from  $1 - \omega$  and from 1, and if  $\mu$ ,  $r$  and  $\omega$  satisfy equation (2.9), then  $\lambda$  is an eigenvalue of  $L_{r,\omega,0}$ .

*Remarks.* (1) Corollary 2.1 coincides with Theorem 1 of [14].

- (2) From Corollary 2.2, we get the necessary condition for the convergence of the MAOR-like method when  $m > n$ , which is: "If the MAOR-like method converges, then

$$(2.10) \quad 0 < \omega < 2."$$

- (3) The relationship between the eigenvalues of the iteration matrices  $L_{r,\omega,\alpha}$  and those of  $-Q^{-1}B^T A^{-1}B$ , obtained in Theorem 2.1, allow us to analyze the convergence of the MAOR-like method and to obtain its convergence regions.
- (4) Corollary 2.3 coincides with Theorem 1 of [11].

(5) As equation (2.5) is a real quadratic equation, in the next section, we will use a result from Young [16] in order to obtain a necessary and sufficient convergence conditions for the MAOR-like method. Therefore, we have:

**Lemma 2.3** (Young [16]). *Both roots of the real quadratic equation  $\lambda^2 - b\lambda + c = 0$ , are less than one in modulus if and only if  $|c| < 1$  and  $|b| < 1 + c$ .*

### 3. CONVERGENCE ANALYSIS

In the sequel, we assume that  $\lambda$  and  $\mu$  are eigenvalues of the matrices  $L_{r,\omega,\alpha}$  and  $-Q^{-1}B^T A^{-1}B$ , respectively, and  $m > n$ . In order to analyze the convergence of the MAOR-like method we have to consider that the parameter  $\omega \in (0, 2)$  which agrees with the second remark after Corollary 2.3. Thus, whether  $1 - \omega$  is an eigenvalue of  $L_{r,\omega,\alpha}$  or not, the convergence property of the MAOR-like method will not be affected. Therefore, the MAOR-like method converges if and only if the spectral radius of  $L_{r,\omega,\alpha}$  is less than one for any eigenvalue  $\mu$ , or equivalently, the modulus of any solution  $\lambda$  of equation (2.5) is less than one. Rewrite equation (2.5) as

$$(3.1) \quad \lambda^2 - b\lambda + c = 0,$$

where

$$(3.2) \quad b = 2 - \omega - \frac{\omega r \mu}{1 - r\alpha} \text{ and } c = (1 - \omega) - \frac{(\omega r - \omega^2)\mu}{1 - r\alpha}.$$

Then, we have the following result:

**Theorem 3.1.** *Assume that the parameter  $\omega$  satisfies (2.10). If the nonsingular matrix  $Q$  is chosen such that all the eigenvalues of the matrix  $-Q^{-1}B^T A^{-1}B$  are real and positive, then the MAOR-like method converges if and only if*

$$(3.3) \quad 1 - r\alpha > 0, \quad 1 - r\alpha - (\omega - r)\mu_{\max} > 0 \quad \text{and} \quad \frac{\omega(2r - \omega)\mu_{\max}}{1 - r\alpha} < 2(2 - \omega),$$

where  $\mu_{\max}$  is the largest eigenvalue of the matrix  $-Q^{-1}B^T A^{-1}B$ .

*Proof.* Let  $S$  be the set of all eigenvalues  $\mu$  of the matrix  $-Q^{-1}B^T A^{-1}B$ . The coefficients of the quadratic equation (3.1) are obviously real, then, from Theorem 2.1 and Lemma 2.3, we can conclude that the MAOR-like method converges if and only if

$$(3.4) \quad \begin{cases} |1 - \omega - \frac{(\omega r - \omega^2)\mu}{1 - r\alpha}| < 1, \\ |2 - \omega - \frac{\omega r \mu}{1 - r\alpha}| < 1 + (1 - \omega) - \frac{(\omega r - \omega^2)\mu}{1 - r\alpha}, \end{cases} \quad \text{for any } \mu \in S,$$

which is equivalent to

$$(3.5) \quad \begin{cases} 1 - r\alpha > 0, \\ 1 - r\alpha - (\omega - r)\mu > 0, \\ \frac{2\omega r - \omega^2}{1 - r\alpha} \mu < 2(2 - \omega), \\ \frac{2\omega(r - \omega)\mu}{1 - r\alpha} < 2(2 - \omega), \end{cases} \quad \text{for any } \mu \in S.$$

Since  $\mu > 0$  for any  $\mu \in S$  and we have proved that  $1 - r\alpha > 0$ , hence, the third and fourth inequalities of (3.5) are equivalent to the last inequality of (3.3), as

$$\max \left\{ \frac{2\omega(r - \omega)\mu_{\max}}{(1 - r\alpha)}, \frac{\omega(2r - \omega)\mu_{\max}}{(1 - r\alpha)} \right\} = \frac{\omega(2r - \omega)\mu_{\max}}{(1 - r\alpha)}.$$

Therefore, we proved that the MAOR-like method converges if and only if (3.3) holds.  $\square$

If we let  $\omega = r$  in Theorem 3.1, we obtain the following result:

**Corollary 3.1.** *Assume that the parameter  $\omega$  satisfies (2.10). If the nonsingular matrix  $Q$  is chosen so that all the eigenvalues of the matrix  $-Q^{-1}B^T A^{-1}B$  are real and positive, then the MSOR-like method converges if and only if*

$$(3.6) \quad 1 - \omega\alpha > 0 \quad \text{and} \quad \frac{\omega^2 \mu_{\max}}{(1 - \omega\alpha)} < 2(2 - \omega),$$

where  $\mu_{\max}$  is the largest eigenvalue of the matrix  $-Q^{-1}B^T A^{-1}B$ .

On the other hand, if we let  $\alpha = 0$  in Theorem 3.1, we obtain the following result:

**Corollary 3.2.** *Assume that the parameter  $\omega$  satisfies (2.10). If the nonsingular matrix  $Q$  is chosen so that all the eigenvalues of the matrix  $-Q^{-1}B^T A^{-1}B$  are real and positive, then the AOR-like method converges if and only if*

$$(3.7) \quad 1 - (\omega - r)\mu_{\max} > 0 \quad \text{and} \quad \omega(2r - \omega)\mu_{\max} < 2(2 - \omega),$$

where  $\mu_{\max}$  is the largest eigenvalue of the matrix  $-Q^{-1}B^T A^{-1}B$ .

*Remarks.* (1) Corollary 3.1 coincides with Theorem 2 of [14].

(2) Corollary 3.2 coincides with Theorem 2 of [11].

**Theorem 3.2.** *Assume that the parameter  $\omega$  satisfies (2.10) and  $\alpha \geq 0$ . If the nonsingular matrix  $Q$  is such that all the eigenvalues of the matrix  $-Q^{-1}B^T A^{-1}B$  are real and positive, then the MAOR-like method converges if and only if*

$$(3.8) \quad 1 - r\alpha > 0, \quad 1 - r\alpha - (\omega - r)\mu_{\max} > 0 \quad \text{and} \quad r < \frac{\omega^2 \mu_{\max} + 4 - 2\omega}{2\omega \mu_{\max} + 4\alpha - 2\omega\alpha},$$

where  $\mu_{\max}$  is the largest eigenvalue of the matrix  $-Q^{-1}B^T A^{-1}B$ .

*Proof.* From the last inequality of (3.3) we have

$$\mu_{\max}\omega(2r - \omega) < 2(2 - \omega)(1 - r\alpha)$$

or

$$(2\omega\mu_{\max} + 4\alpha - 2\omega\alpha)r < -2\omega + 4 + \omega^2\mu_{\max}$$

or, equivalently,

$$r < \frac{\omega^2 \mu_{\max} + 4 - 2\omega}{2\omega \mu_{\max} + 4\alpha - 2\omega\alpha}.$$

Hence, the required result is obtained.  $\square$

#### 4. NUMERICAL RESULTS AND DISCUSSION

We now describe some numerical experiments which were carried out in order to analyze the behaviour of the MAOR-like method for different values of the parameters  $\alpha$ ,  $\omega$  and  $r$  and also to compare this iterative method with the MSOR-like method [14], the AOR-like method [11] and the SOR-like method [4]. The computational study was done for two problems used in literature.

The first example, analyzed in Subsection 4.1, has been chosen from [14]. Here, we consider the linear system of equations (1.1) with the matrix  $Q$  and the stopping criterion, used in the computations, chosen according to [14]. For the second example, we consider the Stokes problem; this problem was discussed in [17]. A comparison of the MAOR-like method with the SSOR-like method of [17] and also with the other methods presented earlier is made. The stopping criterion and the different choices of the matrix  $Q$  were chosen according to [17].

The numerical experiments have been performed using Matlab 7.9, on Core 2 Duo, 2.26 GHZ (4GM RAM), laptop (MacBook Pro) with the Macintosh system. The methods have been compared in terms of the error, number of iterations and CPU time (in seconds).

**4.1. First example.** Let us consider the linear system of equations (1.1), given in [6], where the matrices  $A$  and  $B$  are defined as follows:

$$(4.1) \quad \begin{aligned} A = (a_{i,j})_{m \times m} &= \begin{cases} i+1, & i=j, \\ 1, & |i-j|=1, \\ 0, & \text{otherwise;} \end{cases} \\ B = (b_{i,j})_{m \times n} &= \begin{cases} j, & i=j+m-n, \\ 0, & \text{otherwise.} \end{cases} \end{aligned}$$

The vectors  $p$  and  $q$  are chosen so that the components of the vectors  $x$  and  $y$  of (1.1) have values equal to 1. In [4], different choices for the preconditioning matrix  $Q$  were considered. However, in order to compare the MAOR-like method with the MSOR-like, AOR-like and SOR-like methods, presented in [14], [11] and [4], respectively, we have chosen the preconditioning matrix  $Q = -B^T B$  and the stopping criterion

$$e_1 = \frac{\|r_k\|_2}{\|r_0\|_2} < 10^{-6},$$

where

$$r_k = \begin{pmatrix} p \\ q \end{pmatrix} - \begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix} \text{ and } \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix}$$

is the  $k$ th iterate for each iterative method. The “optimum” values for the parameters of the MAOR-like, AOR-like, MSOR-like and SOR-like methods were obtained by running the program for different values of the parameters and choosing those which give the minimum number of iterations.

In order to analyze the performance of the MAOR-like method and to compare the method to the other methods described in Section 1, numerical experiments were carried out. Different values of the parameter  $\alpha$  in the interval  $[0, 2]$  were considered using steps of length 0.1, for each value of  $\alpha$  we consider a range of values for the two parameters  $\omega$  and  $r$  in the intervals  $(0, 2)$  and  $[0, 2]$ , respectively, with steps of length 0.1. The values used in the experiments for  $m$  and  $n$  of (4.1) were  $m = 50$  and  $n = 40$ . The graphs of the number of iterations for a selected values of  $\alpha$  and the above range of values of  $\omega$  and  $r$  are shown Figures 1 to 6. The evolution of the number of iterations (*iter*) for different values of the parameters  $\omega$  and  $r$  can be seen in these figures. To simplify the visualization of the results, we reduced the maximum number of iterations from 5000 to 1000. The gray region shows the values of  $\omega$  and  $r$  (for the fixed  $\alpha$ ) which are not inside the convergence region, i.e., the values of  $\alpha$ ,  $\omega$  and  $r$  which do not satisfy (3.3). In each figure, we have also indicated the values of the parameters for which the minimum number of iterations (*minIT*) has been reached.

From these figures we can conclude that the minimum number of iterations is obtained when the parameters  $\alpha$ ,  $\omega$  and  $r$  are near the boundary of the convergence region. In some cases, we can see that the number of iterations increases when  $\omega$  and  $r$  are very near the boundary of the convergence region. We would like to point out that we only present here some cases with a big step. However, when we shorten it, later, this fact will be more clarified.

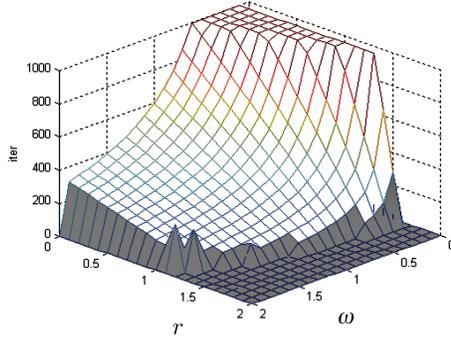


FIGURE 1.  $\alpha = 0.6$ ,  $r = 1.5$ ,  $\omega = 1.0$ ,  $\text{minIT} = 64$ .

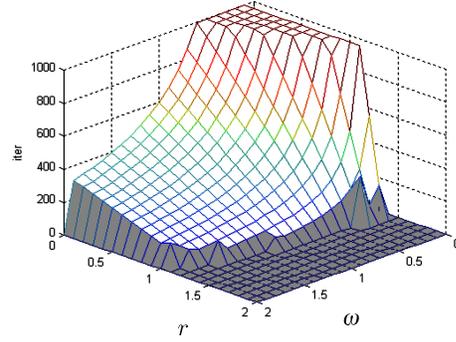


FIGURE 2.  $\alpha = 0.7$ ,  $r = 1.3$ ,  $\omega = 1.0$ ,  $\text{minIT} = 54$ .

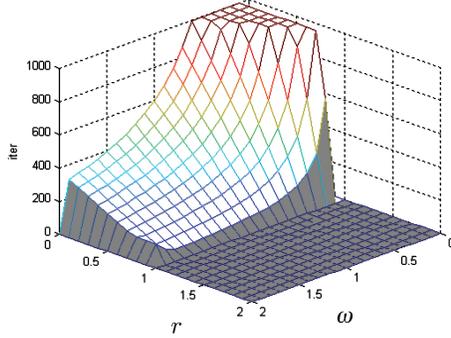


FIGURE 3.  $\alpha = 1.0$ ,  $r = 0.9$ ,  $\omega = 1.6$ ,  $\text{minIT} = 32$ .

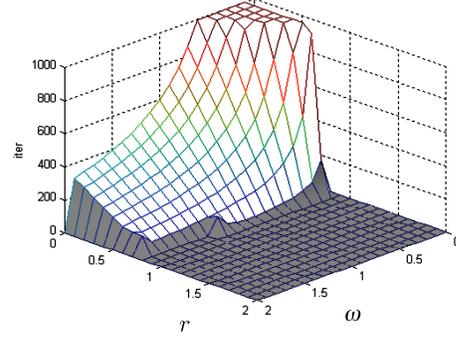


FIGURE 4.  $\alpha = 1.2$ ,  $r = 0.8$ ,  $\omega = 0.9$ ,  $\text{minIT} = 18$ .

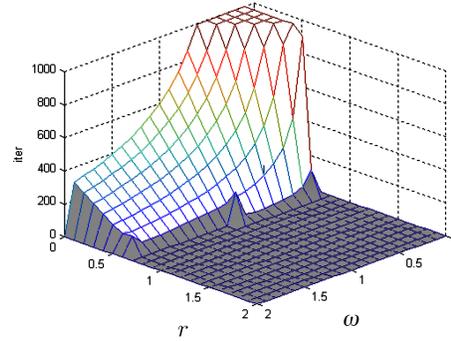


FIGURE 5.  $\alpha = 1.4$ ,  $r = 0.7$ ,  $\omega = 0.7$ ,  $\text{minIT} = 23$ .

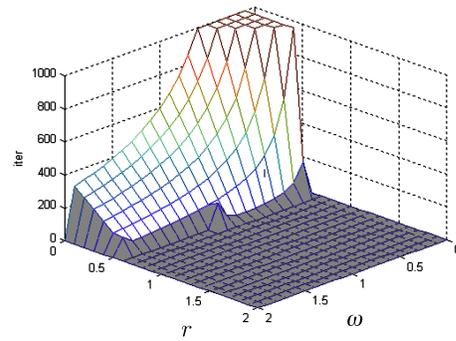


FIGURE 6.  $\alpha = 1.6$ ,  $r = 0.6$ ,  $\omega = 0.7$ ,  $\text{minIT} = 28$ .

Figures 7 and 8 show the graphs of the number of iterations for  $\alpha = 0$  (AOR-like method) and  $\alpha = 0.5$  (MAOR-like method), respectively, and for  $m = 50$  and  $n = 40$ .

In both figures we have drawn a line which corresponds to the number of iterations when  $\omega = r$ . These lines correspond to the results obtained with the SOR-like method (Figure 7) and MSOR-like method (Figure 8), respectively.

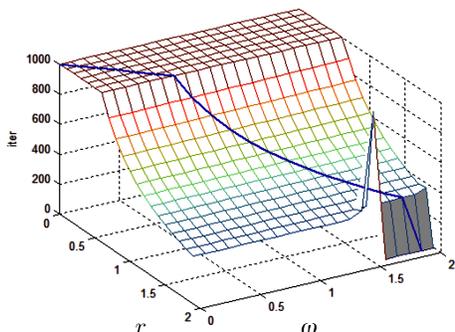


FIGURE 7.  $\alpha = 0$ ,  $r = 0.1$ ,  $\omega = 1.9$ ,  $\text{minIT} = 314$  (AOR-like).  
 $\alpha = 0$ ,  $r = 1.8$ ,  $\omega = 1.8$ ,  $\text{minIT} = 341$  (SOR-like).

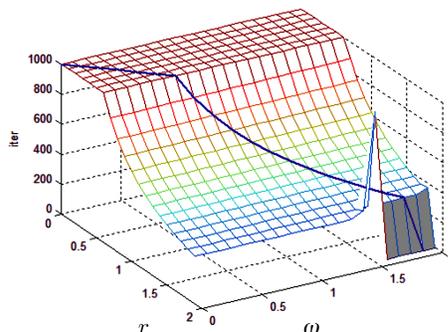


FIGURE 8.  $\alpha = 0.5$ ,  $r = 1.8$ ,  $\omega = 0.8$ ,  $\text{minIT} = 84$  (MAOR-like).  
 $\alpha = 0.5$ ,  $r = 1.5$ ,  $\omega = 1.5$ ,  $\text{minIT} = 101$  (MSOR-like).

These figures reinforce the conclusions we have obtained before. In addition, we can see that the number of iterations for  $\alpha = 0$  is significantly higher than for other values of  $\alpha$ . It can be noted that the number of iterations increases when  $\omega$  and  $r$  are close to the boundary of the convergence region.

The results for the SOR-like ( $\omega = r$  and  $\alpha = 0$ ) and MSOR-like ( $\omega = r$  and  $\alpha \neq 0$ , in this case  $\alpha = 0.5$ ) methods can be seen more clearly in Figure 9. Here, we use a solid line and a dashed line to represent the number of iterations obtained with the SOR-like and the MSOR-like methods, respectively. The gray region has the same meaning as in the previous figures.

From this figure we emphasize the superiority of the MSOR-like method over the SOR-like method. Furthermore, we can conclude that the minimum number of iterations is obtained near the boundary of the convergence region.

A much more detailed analysis, using steps of length 0.01 to generate values of the parameters  $\alpha \in [0, 2]$ ,  $\omega \in (0, 2)$  and  $r \in [0, 2]$ , was carried out, the obtained results shows that the methods are quite sensitive to small changes in the values of the parameters when they are near the boundary of the convergence region. To clarify this conclusion we present Figures 10 and 11 which show the number of iterations when the two parameters  $\omega$  and  $r$  are near the optimal regions for  $\alpha = 0$  and  $\alpha = 0.5$ , respectively.

In these figures we can see that the number of iterations increases suddenly when  $\omega$  and  $r$  take values near the boundary of the convergence region and this boundary appears at the back of the hill. In order to visualize this situation more clearly we show the results of the SOR-like method ( $\alpha = 0$ ) and the MSOR-like method ( $\alpha = 0.5$ ) in Figures 12 and 13, respectively.

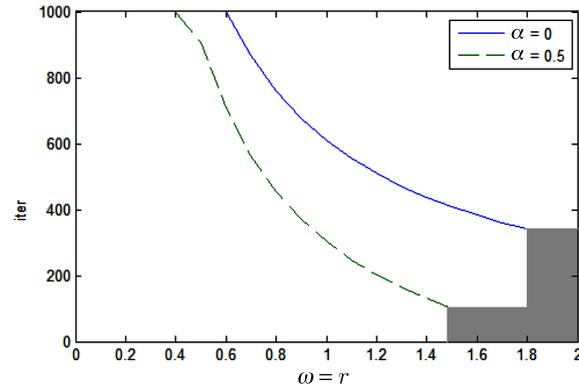
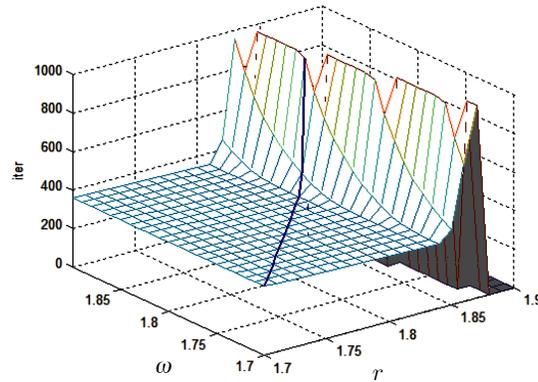


FIGURE 9. Number of iterations when  $\alpha = 0$  (right) and  $\alpha = 0.5$  (left).



$$\alpha = 0, r = 1.70, \omega = 1.84, \text{minIT} = 333 \text{ (AOR-like)}$$

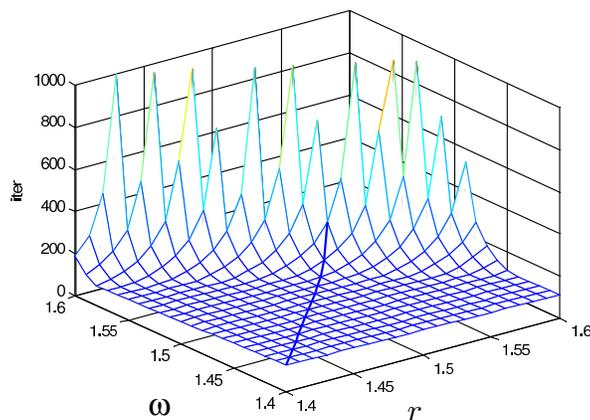
$$\alpha = 0, r = 1.82, \omega = 1.82, \text{minIT} = 337 \text{ (SOR-like)}$$

FIGURE 10. Number of iterations when  $\alpha = 0$  and  $\omega, r$  are near the optimal regions.

The same behaviour can be noticed for the SOR-like method in the graphs obtained by Golub et al. [4] for the spectral radius.

The obtained results for the SOR-like and AOR-like methods are summarized in Table 1 and the results for the MSOR-like and MAOR-like methods are summarized in Table 2. The two tables show the minimum number of iterations and computational time required for different values of  $m$  and  $n$  using the stopping criterion suggested in [14], i.e.,  $e_1 < 10^{-6}$  and number of iterations  $< 5000$ .

From the results obtained and recorded in Tables 1 and 2, we can conclude that the MSOR-like method and MAOR-like method are both superior to the SOR-like method and AOR-like method in terms of number of iterations and computational time. It can also be noted that the restriction  $\omega = r$ , for the SOR-like method and the MSOR-like method, slightly increases the number of iterations in comparison to the versions without this restriction, i.e., the AOR-like method and the MAOR-like



$\alpha = 0.5, r = 1.57, \omega = 1.41, \text{minIT} = 93$  (MAOR-like)  
 $\alpha = 0.5, r = 1.51, \omega = 1.51, \text{minIT} = 100$  (MSOR-like)

FIGURE 11. Number of iterations when  $\alpha = 0.5$  and  $\omega, r$  are near the optimal regions.

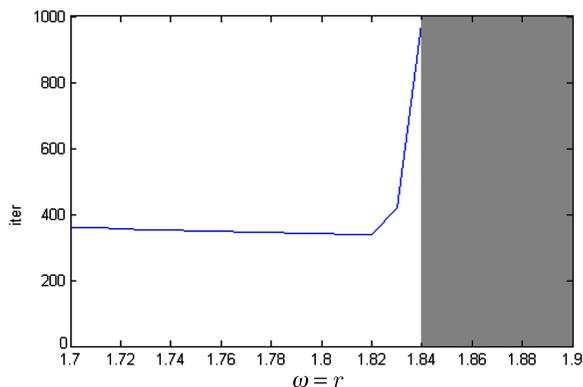


FIGURE 12. Number of iterations when  $\alpha = 0$  and  $\omega = r$  are near the optimal region.

method. However, this augmentation is less than 10% for the SOR-like method and 33% for the MSOR-like method and decreases when the size of the problem increases.

Finally, from Table 2, the optimum values of the parameter  $\alpha$  are close to 1 and when the size of the problem is large then the optimum value of  $\alpha$  is equal to 1 for both MSOR-like and MAOR-like methods. In Table 3 we present the results for the two methods using  $\alpha = 1$ . It can be noted from the results obtained that the number of iterations recorded when  $\alpha = 1$  are very close to those presented in Table 2. Hence, we can use this value as a reference value for  $\alpha$ .

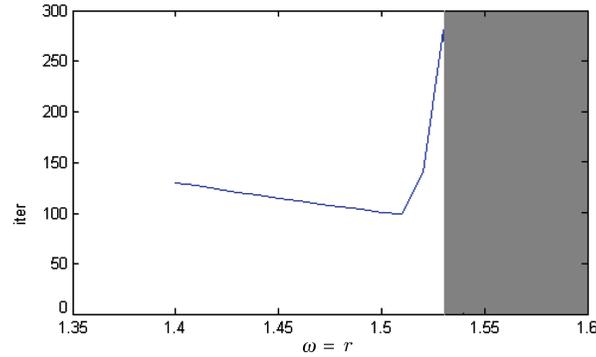


FIGURE 13. Number of iterations when  $\alpha = 0.5$  and  $\omega = r$  are near the optimal region.

TABLE 1. CPU time, iterations number and values of the parameters for the SOR-like and AOR-like methods

Method	$m$	$n$	$\alpha$	$r$	$\omega$	minIT	CPU(s)	$e_1$
SOR-like	50	40	0	1.8201	1.8201	337	0.0537	9.8E-7
	200	150	0	1.9533	1.9533	1201	0.3158	9.9E-7
	400	300	0	1.9759	1.9759	2357	2.7729	9.9E-7
AOR-like	50	40	0	0.0000	1.9522	304	0.0403	9.9E-7
	200	150	0	0.0000	1.9885	1170	0.2631	9.9E-7
	400	300	0	0.0000	1.9935	2326	2.1419	9.9E-7

TABLE 2. CPU time, iterations number and values of the parameters for the MSOR-like and MAOR-like methods

Method	$m$	$n$	$\alpha$	$r$	$\omega$	minIT	CPU(s)	$e_1$
MSOR-like	50	40	1.2	0.8	0.8	20	0.0114	9.9E-7
	200	150	1.00	0.9890	0.9890	21	0.0670	8.3E-7
	400	300	1.00	0.9945	0.9945	21	0.4014	8.5E-7
MAOR-like	50	40	1.12	0.86	0.92	15	0.0089	9.2E-7
	200	150	1.1	0.9	1.0	16	0.0609	8.2E-7
	400	300	1.0	0.9945	0.9945	21	0.4014	8.5E-7

4.2. **Second example.** We consider Stokes problem presented in [17] for the SSOR-like method. Hence, we will find  $\mathbf{u}$  and  $\mathbf{p}$ , such that

$$(4.2) \quad \begin{cases} -\mu\Delta\mathbf{u} + \nabla\mathbf{p} = f, & \text{in } \Omega, \\ \nabla\mathbf{u} = g, & \text{in } \Omega, \\ \mathbf{u} = 0, & \text{on } \partial\Omega, \\ \int_{\Omega} \mathbf{p}(x)dx = 0, \end{cases}$$

with  $\Omega = (0, 1) \times (0, 1) \subset \mathbb{R}^2$ ,  $\Delta$  is the componentwise Laplace operator,  $\partial\Omega$  the boundary of  $\Omega$ ,  $\mathbf{u}$  is the vector-valued function representing the velocity and  $\mathbf{p}$  is the scalar function representing the pressure.

TABLE 3. CPU time, iterations number and values of the parameters for the MSOR-like and MAOR-like methods when  $\alpha = 1.0$

Method	$m$	$n$	$\alpha$	$r$	$\omega$	minIT	CPU(s)	$\epsilon_1$
MSOR-like	50	40	1.00	0.9545	0.9545	24	0.0159	6.9E-7
	200	150	1.00	0.9890	0.9890	21	0.0670	8.3E-7
	400	300	1.00	0.9945	0.9945	21	0.4014	8.5E-7
MAOR-like	50	40	1.00	0.9530	0.9980	23	0.0168	9.9E-7
	200	150	1.00	0.99	0.90	21	0.0718	8.9E-7
	400	300	1.00	0.9945	0.99	21	0.4080	8.4E-7

According to [17], after a suitable discretization, we get the linear system of equations

$$(4.3) \quad \begin{pmatrix} A & B \\ -B^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} f \\ -g \end{pmatrix},$$

where  $f$  and  $g$  are chosen so that the exact solution of the augmented linear system (4.3) is

$$\begin{pmatrix} \mathbf{u}^* \\ \mathbf{p}^* \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \in \mathbb{R}^{m+n}.$$

The matrices  $A$  and  $B$  of (4.3) are defined as follows:

$$(4.4) \quad \begin{aligned} A &= \begin{pmatrix} I \otimes Z + Z \otimes I & 0 \\ 0 & I \otimes Z + Z \otimes I \end{pmatrix} \in \mathbb{R}^{2q^2 \times 2q^2}, \\ B &= \begin{pmatrix} I \otimes V \\ V \otimes I \end{pmatrix} \in \mathbb{R}^{2q^2 \times q^2}, \end{aligned}$$

where  $Z = \frac{1}{h^2} \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{q \times q}$ ,  $V = \frac{1}{h} \text{tridiag}(-1, 1, 0) \in \mathbb{R}^{q \times q}$  with  $h = 1/(q + 1)$  the discretization mesh size and  $\otimes$  the Kronecker product of two matrices.

For a stopping criterion, and in order to compare our results with those presented in [17], we will use the norm of relative error vectors defined as

$$(4.5) \quad e_2 = \frac{\sqrt{\|\mathbf{u}^{(k)} - \mathbf{u}^*\|_2^2 + \|\mathbf{p}^{(k)} - \mathbf{p}^*\|_2^2}}{\|\mathbf{u}^*\|_2 + \|\mathbf{p}^*\|_2}.$$

Furthermore, we consider three different choices for the preconditioning matrix  $Q$  according to [17]:

- Case I:**  $Q = \gamma(B^T B) = -(B^T B)/100$ ,
- Case II:**  $Q = B^T(\beta(\text{diag}(A)^{-1}))B = B^T(-h^2(\text{diag}(A)^{-1}))B$ ,
- Case III:**  $Q = \delta I = -I$ .

We let  $m = 2q^2$  and  $n = q^2$ , the criterion to obtain a suitable approximation of the solution was  $e_2 \leq 10^{-7}$  or  $\text{IT} \geq 50000$  and the initial vector we have used was

$$\begin{pmatrix} \mathbf{u}^{(0)} \\ \mathbf{p}^{(0)} \end{pmatrix} = 0.$$

TABLE 4. Iterations (IT), CPU time and relative error using SSOR-like method

		$m = 242, n = 121$			
Method		$\omega$	minIT	CPU(s)	$e_2$
SSOR- like	Case I	0.6118	100	0.062	9.0318E-8
	Case II	1.0843	203	0.1730	9.5206E-8
	Case III	1.3710	41	0.0310	7.8431E-8

Preliminary results obtained from solving the Stokes problem by the SOR-like and AOR-like methods show that these two methods were less efficient than the MSOR-like and MAOR-like methods. Therefore, for this problem, we will only present computational results for the MSOR-like and MAOR-like methods and we will compare them with the results obtained in [17] for the SSOR-like method.

First, we present the computational results for  $m = 242$  and  $n = 121$ . In Table 4 we summarize the results presented in [17] which were obtained by applying the SSOR-like iterative method to the Stokes problem for the three cases above.

From this table, we can conclude that the preconditioning matrix  $Q$  associated to Case III provides a big reduction in the number of iterations and the system (4.3) can be easily solved. On the other hand, the preconditioning matrix  $Q$  associated to Case II makes the system more difficult to solve since more iterations are needed to obtain the same error.

The numerical results for the MSOR-like and the MAOR-like methods are presented in Table 5, the experimental optimum values of the parameters  $\alpha$ ,  $\omega$  and  $r$  are determined to approximate the solution of the Stokes problem for a range of values of these parameters and choosing those which give the minimum number of iterations (IT), the CPU time and the error  $e_2$  defined for this example were also recorded.

The CPU times, presented in Tables 4 and 5, are not comparable, since the computer used to obtain the results presented in Table 4 is different from the computer used to obtain the results presented in Table 5. However, if we base our comparison on the number of iterations required to solve the problem, then we can conclude that the MSOR-like and MAOR-like methods offer significant economies over the SSOR-like method of [17]. In comparison a savings of between 30% and 50% was achieved. In addition, as in the first example, it can be noted from Table 5 that the MAOR-like method appears to be slightly more efficient than the MSOR-like method.

Furthermore, similar to the results obtained in Table 3 of Subsection 4.1, we let  $\alpha = 1$  and calculate the experimental optimum values of  $r$  and  $\omega$  which gives the minimum number of iterations and the corresponding computational time, the results obtained are recorded in Table 6 for both the MSOR-like and MAOR-like methods. It can be noted that the two methods are still more efficient than the SSOR-like method in terms of the number of iterations.

Next, we present some numerical results for  $m = 2048$  and  $n = 1024$ , using the same stopping criterion, i.e.,  $e_2 \leq 10^{-7}$  or  $IT \geq 50000$ . In Table 7, we summarize the results obtained in [17] to approximate the solution of the Stokes problem using the SSOR-like method for the three different cases where the preconditioned matrix  $Q$  is chosen.

TABLE 5. Iterations (IT), CPU time and relative error for the second example using MSOR-like and MAOR-like methods

		$m = 242, n = 121$					
Method		$\alpha$	$r$	$\omega$	minIT	CPU(s)	$e_2$
MSOR-like	Case I	1.3963	0.4815	0.4815	63	0.0747	9.5E-8
	Case II	1.5815	0.6173	0.6173	95	0.0880	9.5E-8
	Case III	0.9926	0.7444	0.7444	28	0.0259	7.7E-8
MAOR-like	Case I	1.4889	0.4667	0.4556	57	0.0729	9.9E-8
	Case II	1.5642	0.62346	0.68025	90	0.0838	9.1E-8
	Case III	1.1111	0.7083	0.6667	28	0.0257	7.7E-8

TABLE 6. Iterations (IT), CPU time and relative error, for the second example, using the MSOR-like and MAOR-like methods

		$m = 242, n = 121$					
Method		$\alpha$	$r$	$\omega$	minIT	CPU(s)	$e_2$
MSOR-like	Case I	1.00	0.5469	0.5469	80	0.0794	9.3E-8
	Case II	1.00	0.9296	0.9296	182	0.1167	9.1E-8
	Case III	1.00	0.7333	0.7333	29	0.0269	6.7E-8
MAOR-like	Case I	1.00	0.5457	0.5481	80	0.0799	9.4E-8
	Case II	1.00	0.9136	1.3222	156	0.1079	9.8E-8
	Case III	1.00	0.7333	0.7333	29	0.0269	6.7E-8

TABLE 7. Iterations (IT), CPU time and relative error, for the second example, using the SSOR-like method

		$m = 2048, n = 1024$				
Method		$\omega$	minIT	CPU(s)	$e_2$	
SSOR-like	Case I	0.6118	678	80.0160	9.8787E-8	
	Case II	1.0843	1399	158.187	9.9427E-8	
	Case III	1.3710	125	14.3590	9.6815E-8	

As expected, the CPU time and the IT value increase with the size of the problem solved. For this reason, we apply the MSOR-like and MAOR-like methods to the same problem with  $m = 2048$  and  $n = 1024$ , in a similar manner to Table 6, we let  $\alpha = 1$ . Then we calculate the experimental optimum values of  $r$  and  $\omega$ , the corresponding minimum number of iterations and the computational time. The results obtained are recorded in Table 8.

By comparing the two tables, it can be noted that the two methods are again more efficient than the SSOR-like method even with  $\alpha = 1$ .

TABLE 8. Iterations (IT), CPU time and relative error, for the second example, using MSOR-like and MAOR-like methods

		$m = 2048, n = 1024$					
Method		$\alpha$	$r$	$\omega$	minIT	CPU(s)	$e_2$
MSOR-like	Case I	1.00	0.5498	0.5498	582	25.40	9.9E-8
	Case II	1.00	0.9333	0.9333	1182	38.12	9.9E-8
	Case III	1.00	0.7567	0.7567	92	5.80	9.9E-8
MAOR-like	Case I	1.00	0.5663	0.5494	561	25.16	9.9E-8
	Case II	1.00	0.9198	1.3481	984	33.41	9.9E-8
	Case III	1.00	0.75	0.87	81	5.48	9.9E-8

## 5. CONCLUDING REMARKS

In this paper a variant of the Accelerated Overrelaxation (AOR) iterative method, denoted by the modified AOR-like (MAOR-like) method for solving the augmented systems has been presented. The convergence analysis of the method has been done and several numerical examples were given. From the obtained results it can be noted that the MSOR-like and MAOR-like methods performed very well, with the MAOR-like method being the most efficient one.

The optimum values of the parameters have been obtained computationally, the results confirm the theoretical analysis presented in Sections 2 and 3. However, further study is needed to determine the theoretical values of these optimum parameters. This subject will be a matter of further research.

## REFERENCES

- [1] Å. Björck, *Numerical stability of methods for solving augmented systems*, in: Proceedings of Recent Developments in Optimization Theory and Nonlinear Analysis, Jerusalem, 1995, Y. Censor and S. Reich, eds., Contemp. Math., **204**, Amer. Math. Soc., Providence, RI (1997), 51-60. MR1442994 (97m:65055)
- [2] H. Elman and G. H. Golub, *Inexact and preconditioned Uzawa algorithms for saddle point problems*, SIAM J. Numer. Anal., **31** (1994) 1645-1661. MR1302679 (95f:65065)
- [3] B. Fischer, A. Ramage, D. J. Silvester and A. J. Wathen, *Minimum residual methods for augmented systems*, BIT **38** (1998) 527-543. MR1652781 (99i:65031)
- [4] G. H. Golub, X. Wu, and Jin-Yun Yuan, *SOR-like methods for augmented systems*, BIT **41** (2001) 71-85. MR1829662 (2002a:65055)
- [5] A. Hadjidimos, *Accelerated overrelaxation method*, Math. Comp. **32** (1978) 149-157. MR0483340 (58:3353)
- [6] Q. Hu and J. Zou, *An iterative method with variable relaxation parameters for saddle-point problems*, Siam J. Matrix Anal. Appl. **23** (2001) 317-338. MR1871315 (2002j:65041)
- [7] C. Li and D. J. Evans, *A new iterative method for large sparse saddle point problem*, Int. J. of Comp. Maths. **74** (2000) 529-536. MR1774384 (2001c:65052)
- [8] C. Li, B. Li and D. J. Evans, *A generalized successive overrelaxation method for least squares problems*, BIT **38** (1998) 347-355. MR1638120 (99e:65066)
- [9] C. Li, B. Li and D. J. Evans, *Optimum accelerated parameter for the GSOR method*, Neural, Parallel & Scientific Computations **7** (1999) 453-462.
- [10] C. Li, Z. Li, X. Shao, Y. Nie, and D. J. Evans, *Optimum parameter for the SOR-like method for augmented systems*, Int. J. of Comp. Maths. **81** (2004) 749-763. MR2170345
- [11] C. Li, Z. Li, Y. Nie, and D. J. Evans, *Generalized AOR methods for the augmented system*, Int. J. of Comp. Maths. **81** (2004) 495-504. MR2170897
- [12] M. M. Martins, *Extended Convergence Regions for the AOR Method*, Journal of Computational and Applied Mathematics, **24** (1988) 327-336. MR974021 (90a:65101)

- [13] M. M. Martins, D. J. Evans and M. E. Trigo, *The AOR iterative method for new preconditioned linear systems*, Journal of Computational and Applied Mathematics, **132** (2001) 461-466. MR1840641
- [14] X. Shao, Z. Li and C. Li, *Modified SOR-like methods for the augmented system*, Int. J. of Comp. Maths. **84** (2007) 1653-1662. MR2376720
- [15] S. Wright, *Stability of augmented system factorization in interior-point methods*, SIAM J. Matrix Anal. Appl. **18** (1997) 191-222. MR1428208 (97i:90039)
- [16] D.M. Young, *Iterative solutions of large linear systems*, Academic Press, NY (1971). MR0305568 (46:4698)
- [17] B. Zheng, K. Wang and Y. Wu , *SSOR-like methods for saddle point problems*, Int. J. of Comp. Maths. **86** (2009) 1405-1423. MR2547562

DEPARTMENT OF MATHEMATICS, UNIVERSITY OF COIMBRA, APARTADO 3008, EC UNIVERSIDADE, 3001-454 COIMBRA, PORTUGAL

*Current address:* Institute of Telecommunications, University of Coimbra, 3000 Coimbra, Portugal

*E-mail address:* `mmartins@mat.uc.pt`

DEPARTMENT OF COMPUTER SCIENCE, LOUGHBOROUGH UNIVERSITY, LOUGHBOROUGH, LE11 3TU, LEICESTERSHIRE, UNITED KINGDOM

*E-mail address:* `W.Yousif@lboro.ac.uk`

DEPARTMENT OF MATHEMATICS, UNIVERSITY OF COIMBRA, APARTADO 3008, EC UNIVERSIDADE, 3001-454 COIMBRA, PORTUGAL

*Current address:* CMUC, Department of Mathematics, University of Coimbra, 3001-454 Coimbra, Portugal

*E-mail address:* `zeluis@mat.uc.pt`